Advanced Java Programming

Week 5 Topic Outline

Data Structures

1. What is a data structure?

    a. An implementation of an abstract data type
    b. Implementations rely on only 2 operations:

        1. Contiguous memory allocation
        2. Memory references (pointers).

2. How can we implement a list?

    a. Mutable array
    b. Linked list (singly linked, doubly linked)

3. How can we implement a stack?

    a. Singly-linked nodes pointing from top to bottom.
    b. External pointer to top node.

4. How can we implement a queue?

    a. Singly-linked nodes pointing from head to tail.
    b. External pointers to both head and tail.
    c. Insert at tail, remove from head.

5. How can we implement a map?

    a. Hash table where hash function yields index and value is stored in array.

6. How can we implement a bounded set?

    a. Boolean array (can also bit-pack)

7. How can we implement an unbounded set?

    a. Naive implementation: use a list. Requires $O(n)$ for basic `insert`, `remove`, `contains` operations.
    b. Use a hash table, storing the element itself.

8. How can we implement a sorted set?

      a. Linked list + Hash table if we only care about insertion order

      b. Binary search tree if we care about sorted order

9. Ponder: how can we implement a tuple? Not so easy...

Java's Collection Framework

Iterable
  Collection
    AbstractCollection
      List
        AbstractList
          ArrayList
          Stack
        AbstractSequentialList
          LinkedList
      Queue
        AbstractQueue
          PriorityQueue
        Deque
          ArrayDeque
          LinkedList

```
        AbstractSet
        EnumSet
        HashSet
        LinkedHashSet

        TreeSet
```

Hashing

--> Talk about hashcodes and overriding hashcode / equals methods